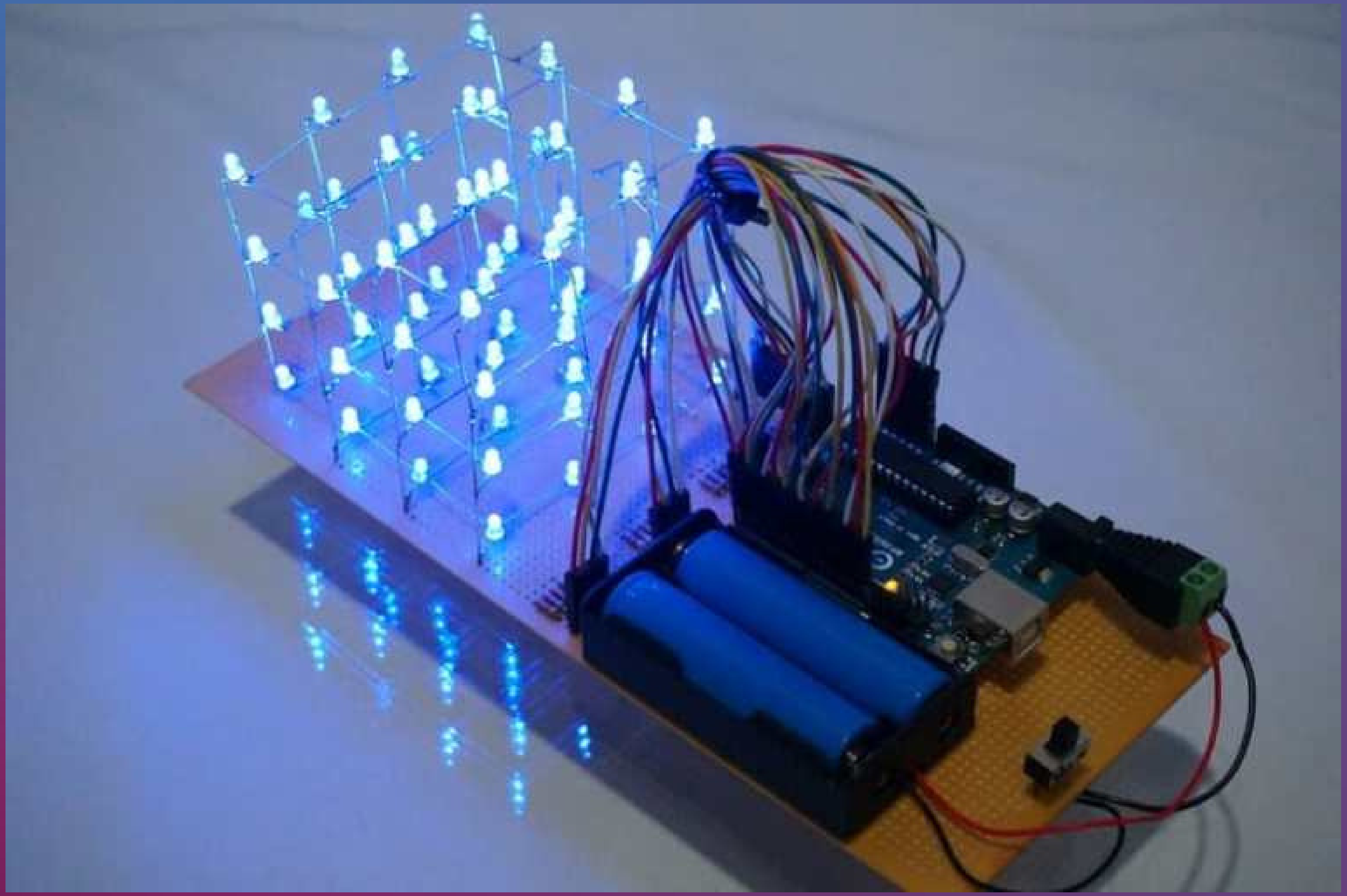
Introduction to

ARDUINO

What it is and how do we use it?

What is Arduino

- Arduino is an open-source electronics platform based on easy-to-use hardware and software.
- Arduino Boards can read inputs from sensor and use them for computational purposes or to command actuators



Arduino Software

- Simulator: WOKWI
- Main function of Arduino
- Useful functions of Arduino
- Digital vs Analog signals
- Arduino Libraries

Introduction to WOKWI Simulator

Text Editor:

This is where you write your code for the Arduino

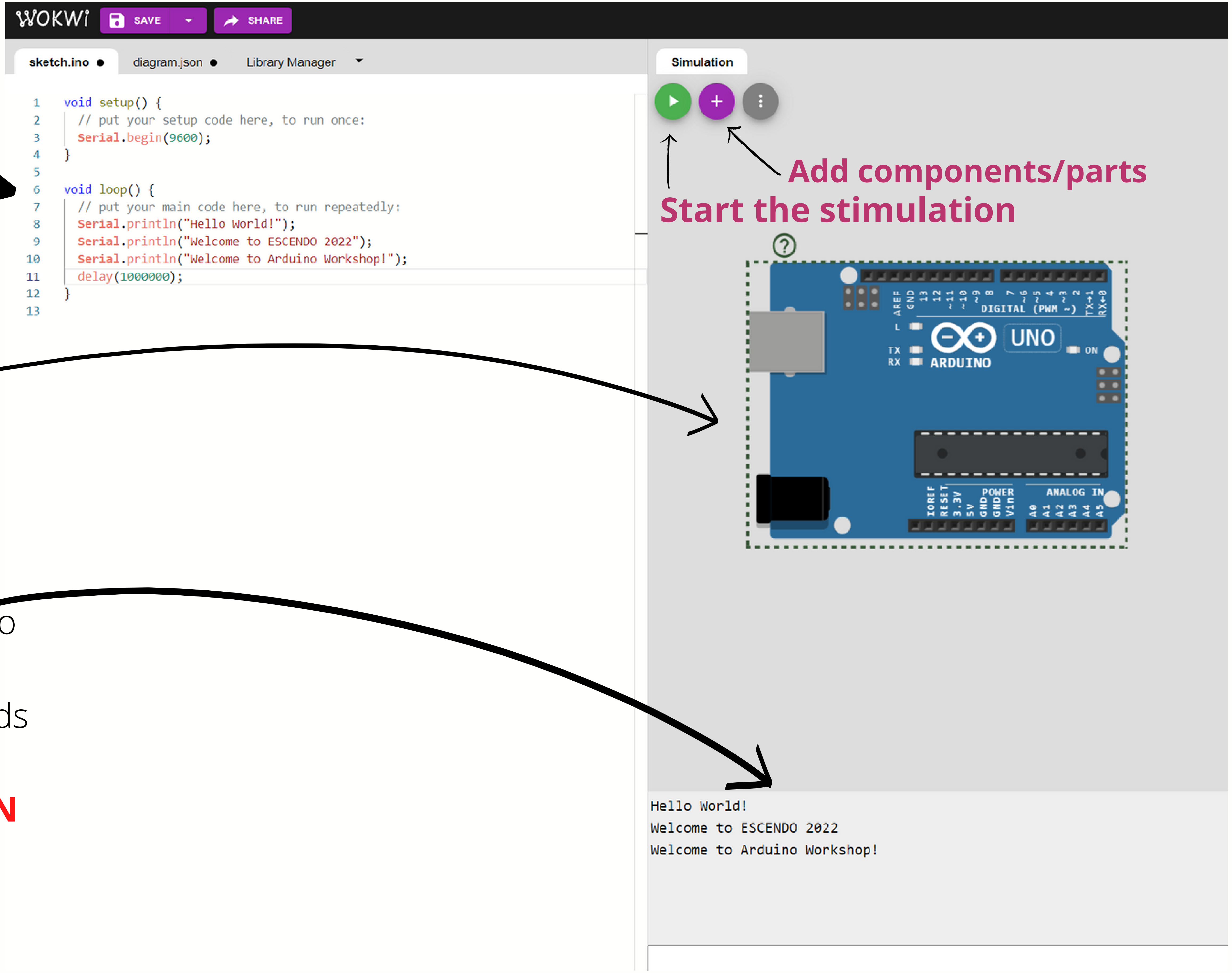
Simulation:

This is where you let WOKWI stimulate your Arduino code

Serial Monitor:

The Serial Monitor provides a way to send/receive information to/from your Arduino code. You can use it to view debug messages printed by your program, or to send commands that control your program.

(ALWAYS TEST YOUR CODE IN SIMULATION BEFORE RUNNING IT ON THE ACTUAL BOARD) It will save you time and effort, I promise ;)



Save your current work

Add components/parts

Start the stimulation

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
}

void loop() {
  // put your main code here, to run repeatedly:
  Serial.println("Hello World!");
  Serial.println("Welcome to ESCENDO 2022");
  Serial.println("Welcome to Arduino Workshop!");
  delay(1000000);
}
```

```
Hello World!
Welcome to ESCENDO 2022
Welcome to Arduino Workshop!
```

Main functions of Arduino

- void setup():
- void loop():

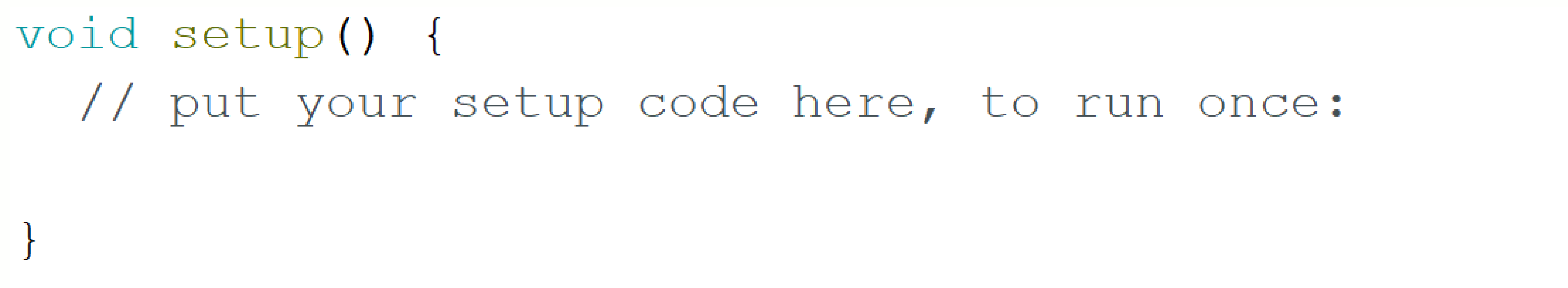
void setup()

- Used to set up your Arduino Code. When the program starts it will run the code in void setup() once.

- Only one void setup() function per source code.

- This is where you put your pinMode function (assigning I/O pins) usually

- Not a good place to put variables in here as the values will not get passed on to the void loop() function.

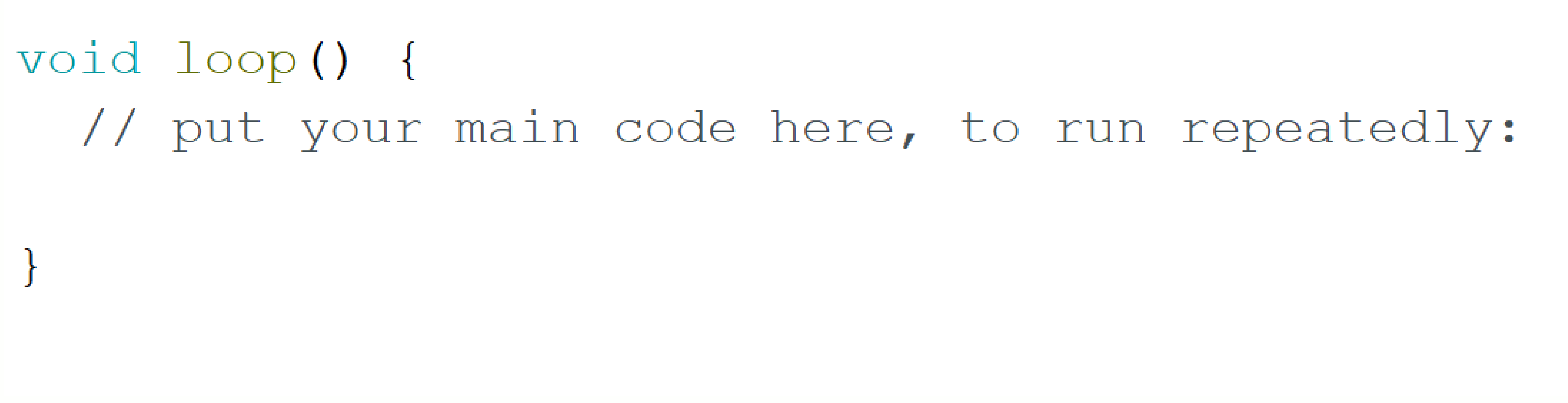


```
void setup() {
  // put your setup code here, to run once:

}
```

void loop()

- Used to loop/repeat your Arduino Code. When the program starts it will run the code in void loop() as long as the Maker Board is turned on.
- Only one void loop() function per source code.
- This is where you put your main source code on what you want the Arduino to do.



```
void loop() {
  // put your main code here, to run repeatedly:

}
```

Useful Arduino functions

More can be found in:
https://www.arduino.cc/reference/en/

1. pinMode(): Set the pins as input or output.
2. digitalWrite()/analogWrite(): Write the output value to the pin. For digital is HIGH or LOW and for analog is 0 to 255.
3. digitalRead()/analogRead(): Read the input value of the pin. For digital it returns a value of HIGH or LOW whereas analog is 0 to 255.
4. delay(): pause the program for x number 0f milliseconds
5. map(): Re-maps a number from one range to another.
6. random(): Randomly generate a number
   randomSeed(): Randomly generate a number based on a seed.
7. Serial.begin(): Sets the data rate in bits per second (baud) for serial data transmission. For communicating with Serial Monitor, make sure to use one of the baud rates listed in the menu at the bottom right corner of its screen. (usually it is Serial.begin(9600). You can set other baud rate as well

Digital Signals

- Discrete signal
- Uses 0 and 1 to represent information
- E.g. a button produces digital signals as it has only 2 states (Pressed/Not Pressed) with each state corresponding to a value (0 or 1), hence we can know the state of the button based on its digital signal

Analog Signals

- Continuous signal
- Uses a range of values to represent information (For Arduino is 0 to 255)
- E.g. a potentiometer produces analog signals as it has a range of values depending on the position of its knob, hence we can obtain the value of the potentiometer by taking its analog signal and use it as information

Libraries

Files that contain functions and data that can be useful when working with specific hardware or writing certain code.

- **Provides extra functionality**

  e.g. working with hardware or manipulating data

- **Download existing library or create your own**

  Sketch > Import Library

- **Some examples**

  Servo.h - for controlling servo motors
  LiquidCrystal.h - for controlling LCDs (will be used)
  pitches.h: Library for the pitch of a note (used later)

Arduino Hardware

- Arduino UNO board
- Breadboard
- LED
- Resistors
- Sensors
- Actuators

Introduction to the Arduino Board

1

Power USB

Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection.

2

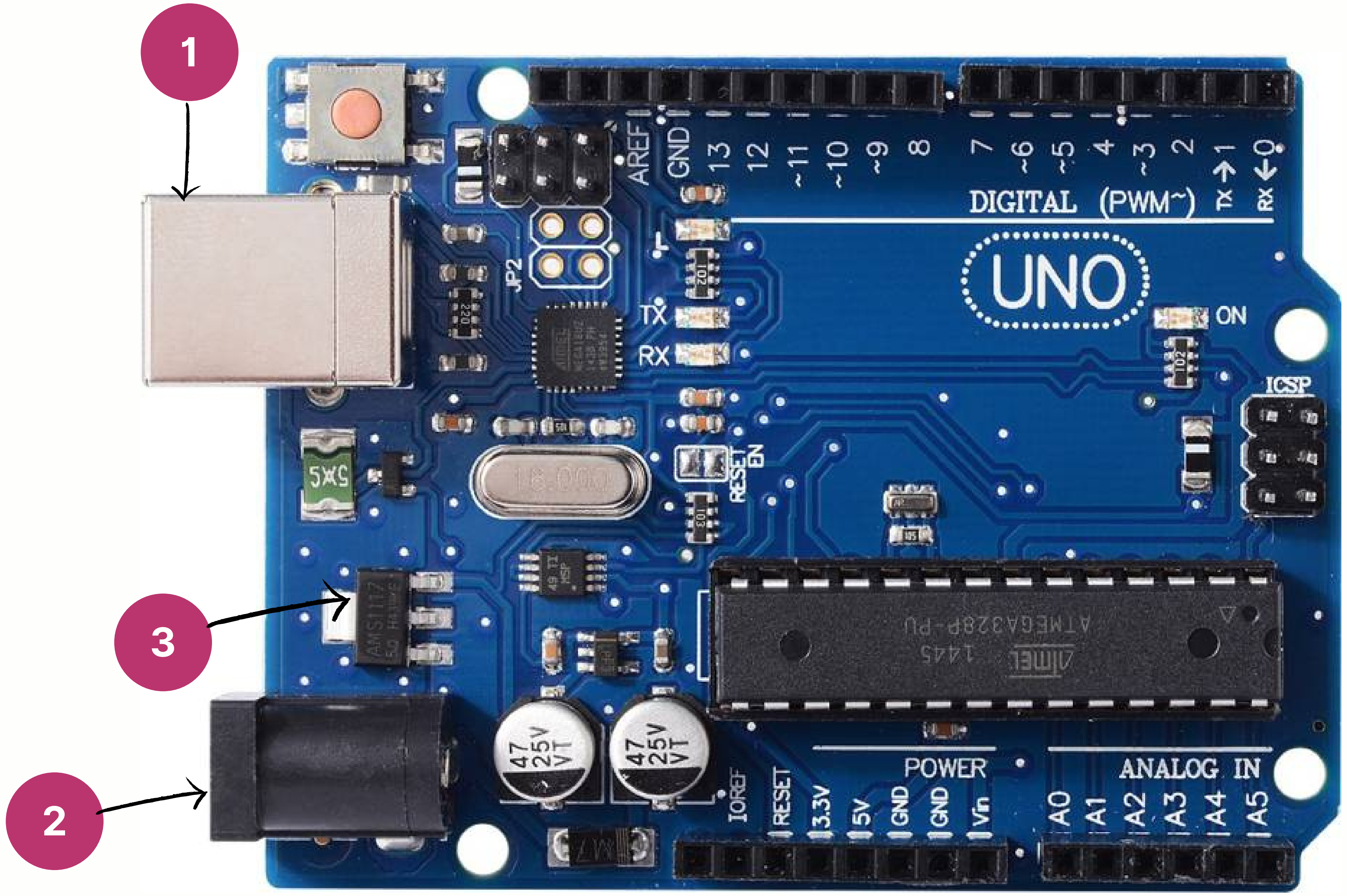
Power (Barrel Jack)

Arduino boards can be powered directly from the AC mains power supply by connecting it to a AC to DC Adapter with Barrel Jack connector

3

Voltage Regulator

The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.



Introduction to the Arduino Board

4

Crystal Oscillator

The crystal oscillator is the clock of the Arduino Board. Its frequency is 16 Mhz (16 Megahetz)

5

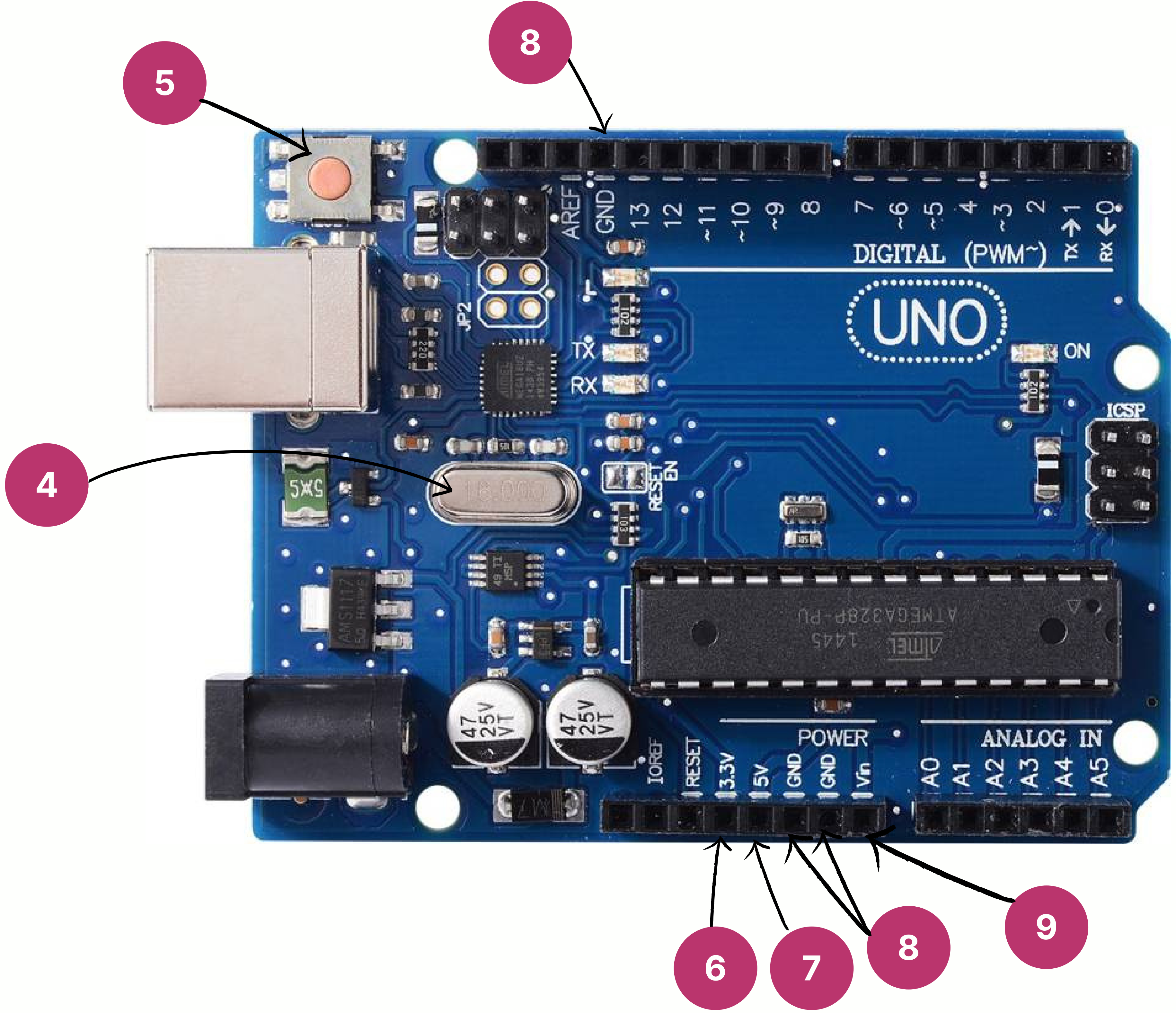
Arduino Reset Button

Use this to reset your Arduino board, i.e., start your program from the beginning.

6 7 8 9

GND,3.3V,5V,Vin

- 3.3V (6) – Supply 3.3 volts output
- 5V (7) – Supply 5 volts output
- GND (8) - Use to ground your circuits (*Always remember to Ground!*)
- Vin (9) – This pin also can be used to power the Arduino board from an external DC power source



Introduction to the Arduino Board

10

Analog pins

The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be processed by the microprocessor.

11

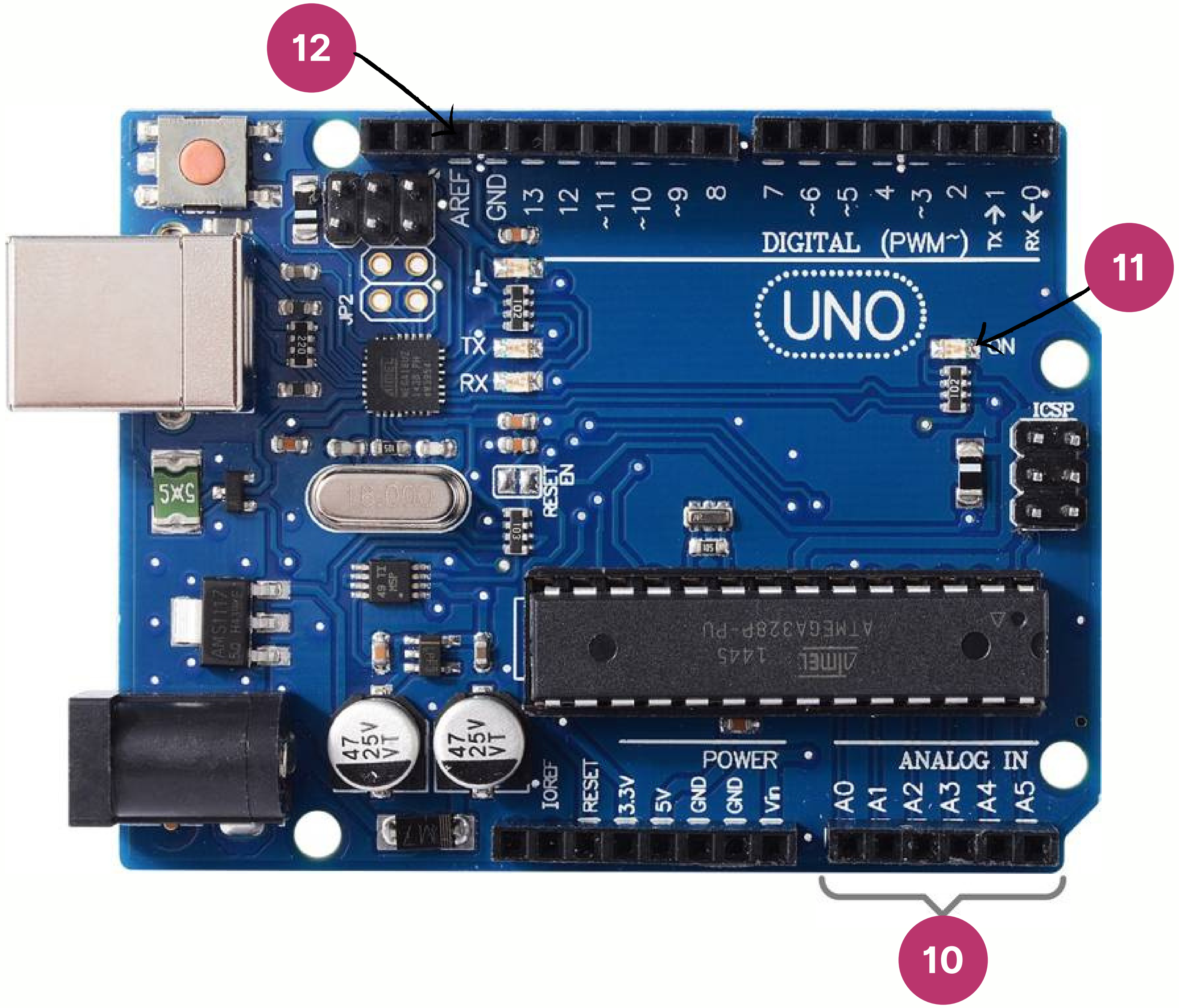
Power LED indicator

Indicates whether your Arduino Board is powered up properly

12

AREF

AREF stands for Analog Reference. It is sometimes, used to set an external reference voltage (between 0 and 5 Volts) as the upper limit for the analog input pins.

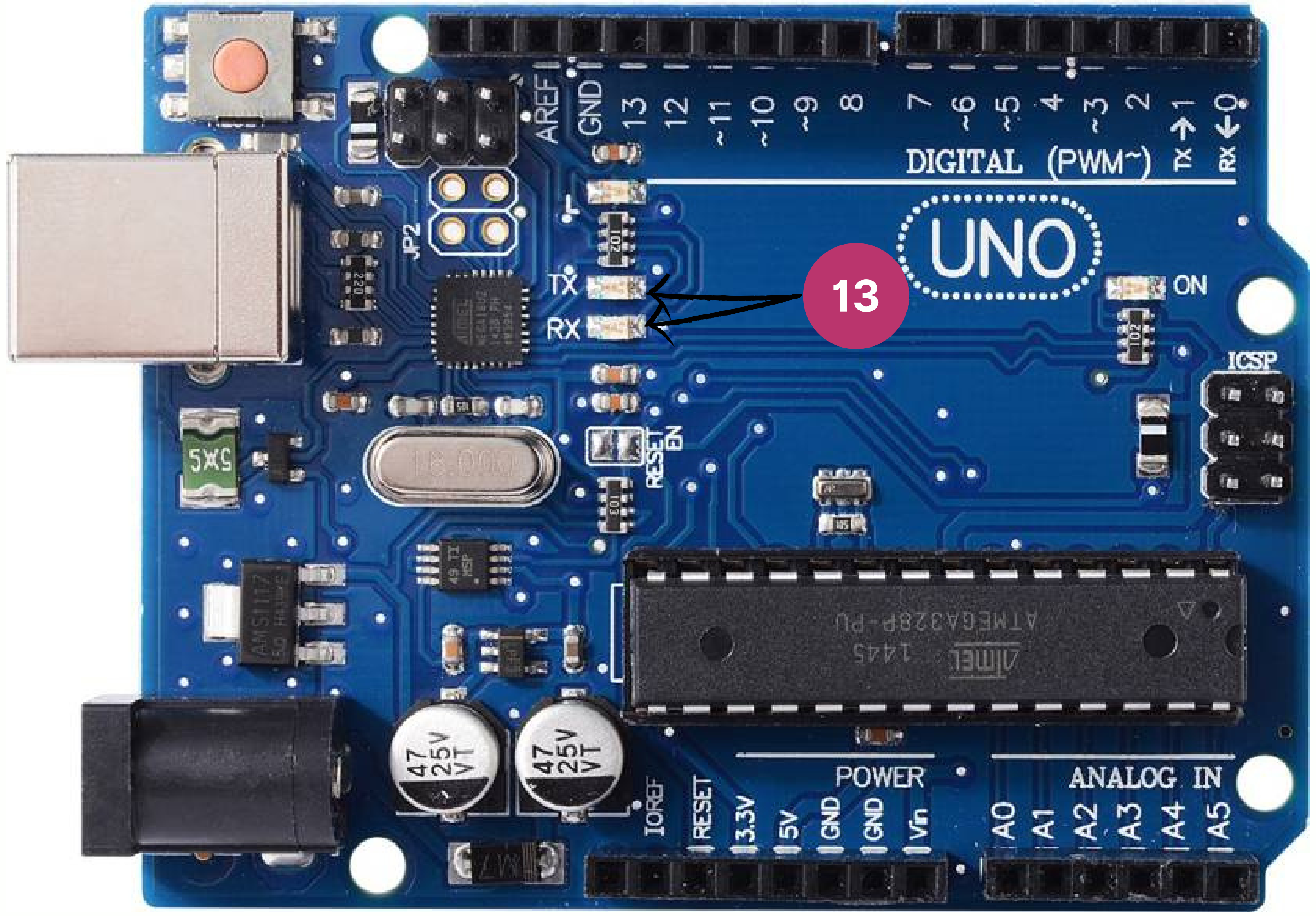


Introduction to the Arduino Board

13 TX/RX

TX and RX LEDs

On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial (Serial Monitor) communication. Second, the TX and RX LED. The TX led flashes with different speed while sending the serial data. The speed of flashing depends on the baud rate used by the board. RX flashes during the receiving process.



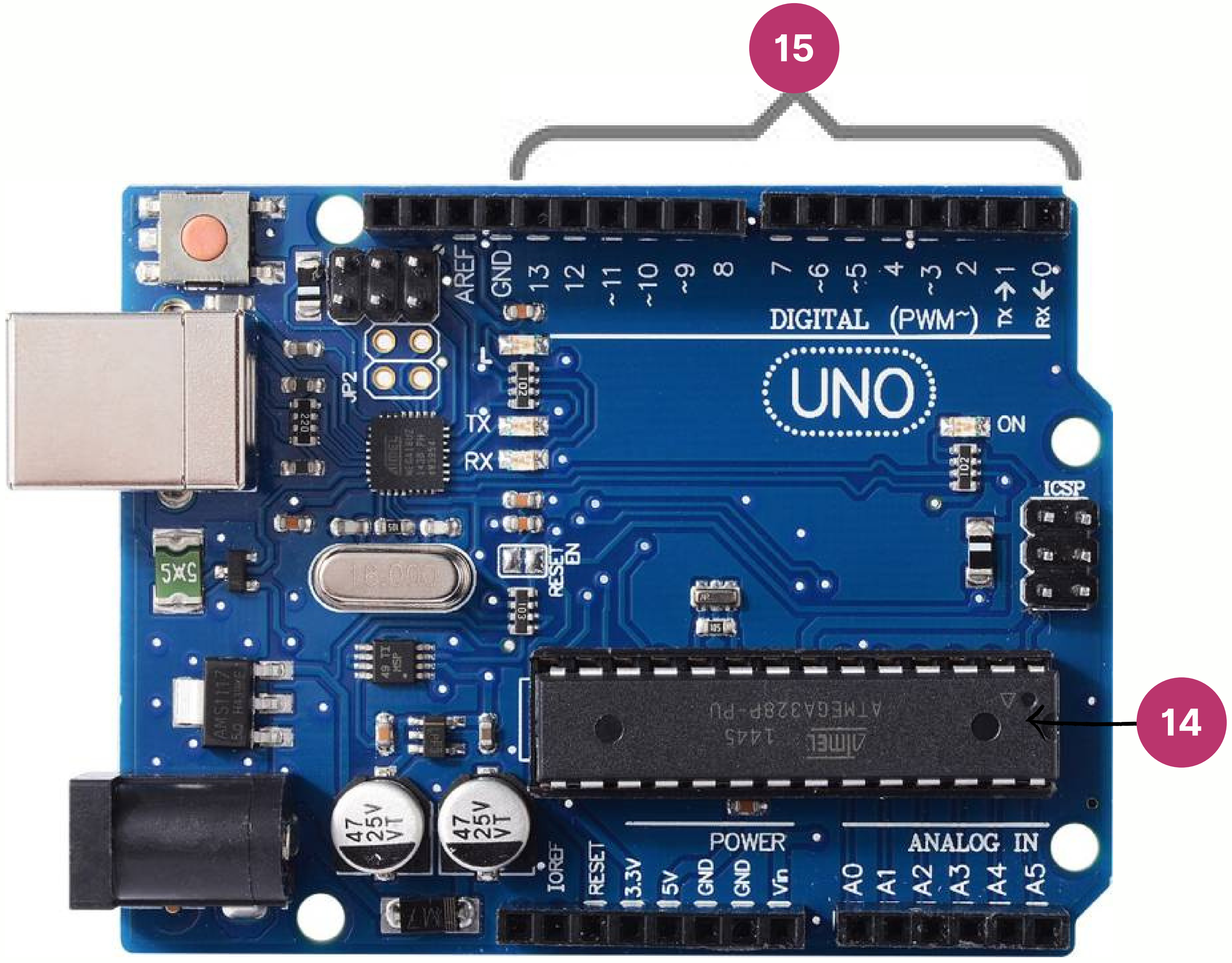
Introduction to the Arduino Board

14 Main Microcontroller

Each Arduino board has its own microcontroller. You can thinkof it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different for each type of board. You must know what IC your board has before loading up a new program from the Arduino IDE. For Arduino UNO, it is ATmega328P.

15 Digital Pins

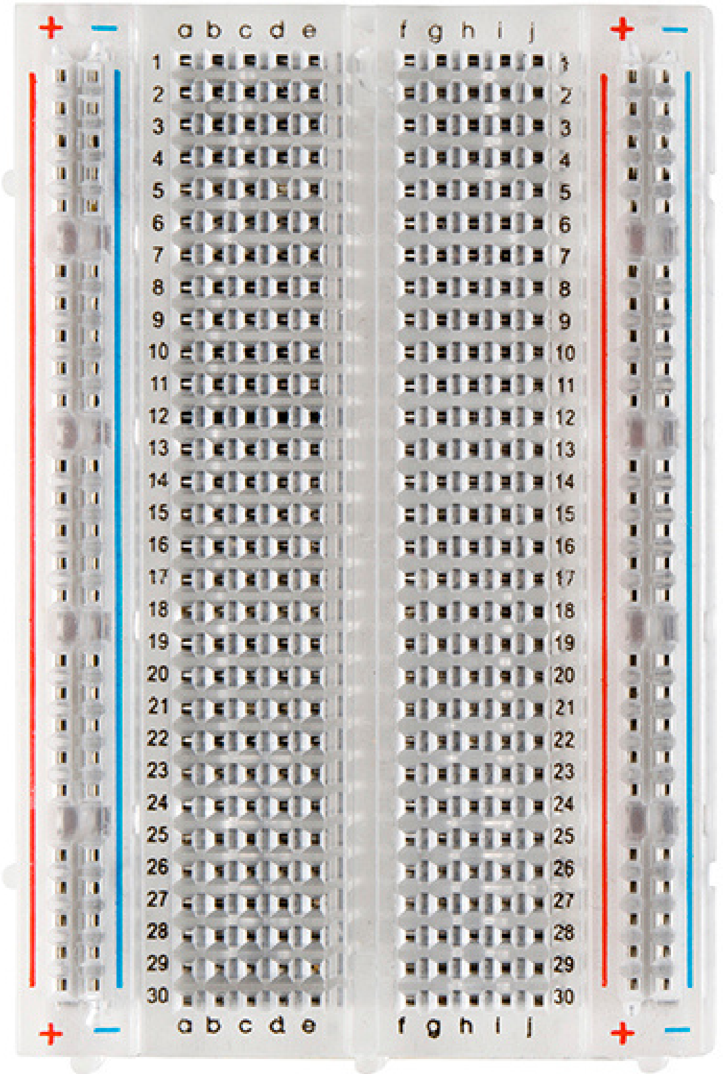
The Arduino UNO board has 14 digital I/O pins (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled “~” can be used to generate PWM.

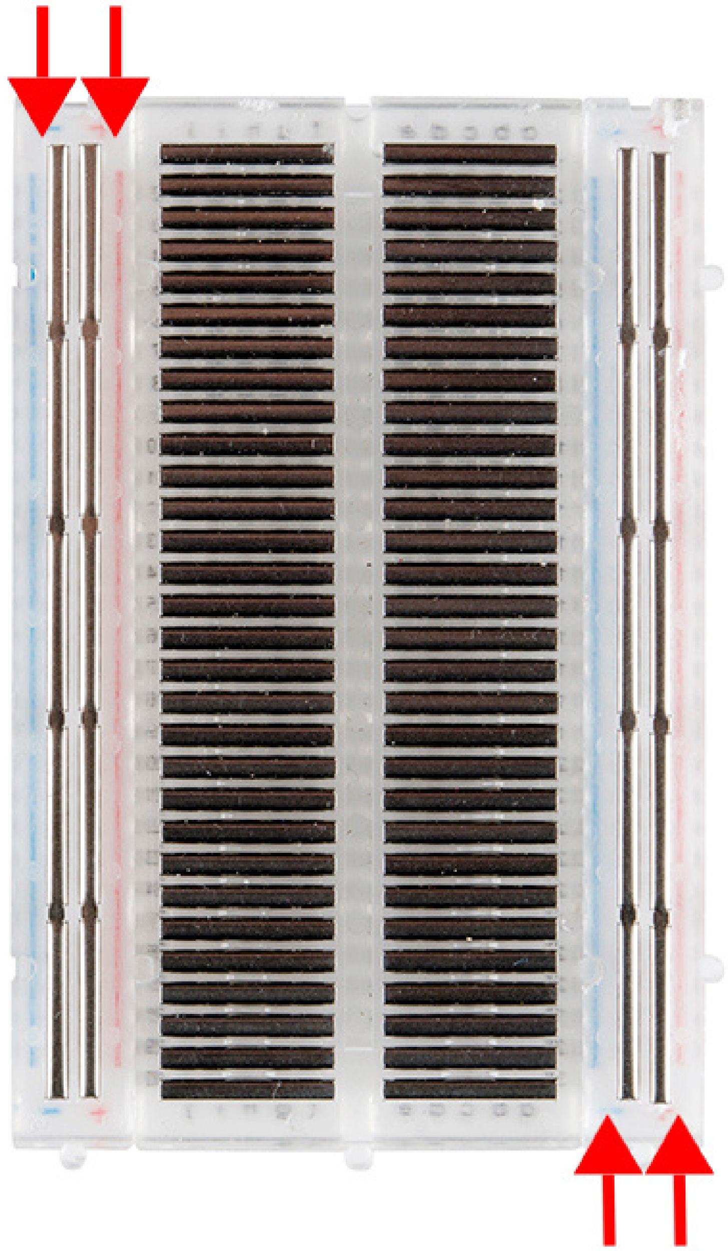


Arduino Board Specs

Explore more!

- **MICROCONTROLLER**: ATmega328P
- **OPERATING VOLTAGE**: 5V
- **INPUT VOLTAGE (RECOMMENDED)**: 7-12V
- **INPUT VOLTAGE (LIMIT)**: 6-20V
- **NO. OF DIGITAL I/O PINS**: 14 (of which 6 provide PWM output)
- **NO. OF PWM DIGITAL I/O PINS**: 6
- **NO. OFANALOG INPUT PINS**: 6
- **DC CURRENT PER I/O PIN**: 20 mA
- **DC CURRENT FOR 3.3V PIN**: 50 mA
- **FLASH MEMORY**: 32 KB (ATmega328P) of which 0.5 KB used by bootloader
- **SRAM**: 2 KB (ATmega328P)
- **EEPROM**: 1 KB (ATmega328P)
- **CLOCK SPEED**: 16 MHz

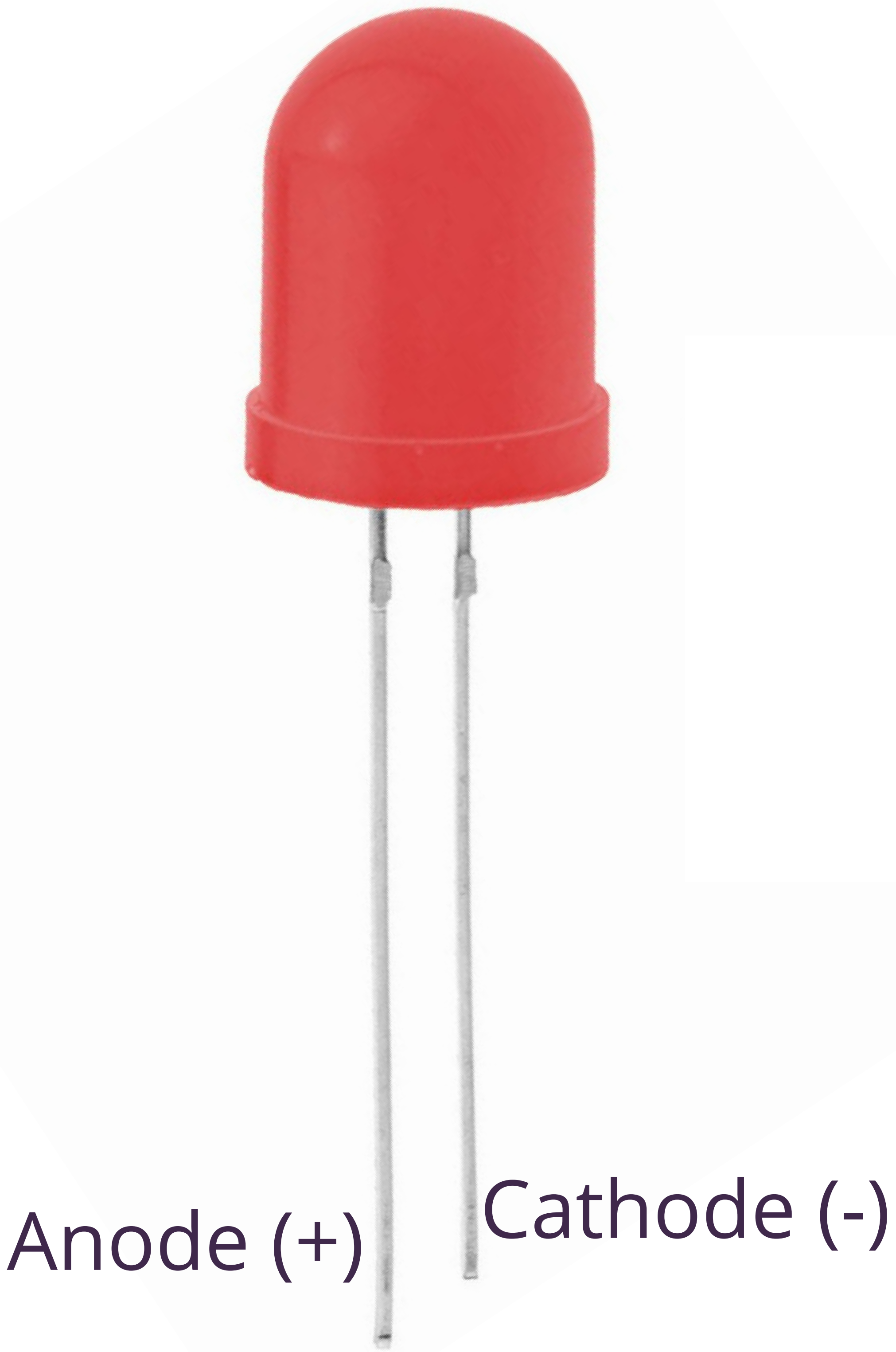


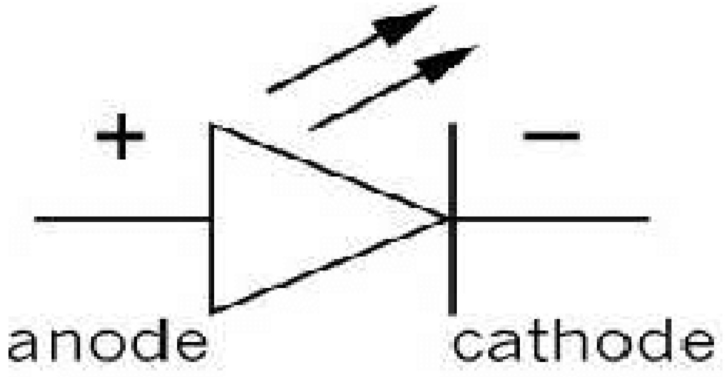


Breadboard

- They are not for you to put your bread on.
- The electronics breadboard is your circuit builder.
- Breadboards are good as it is beginner-friendly, require no soldering and it is easier to use for prototyping.

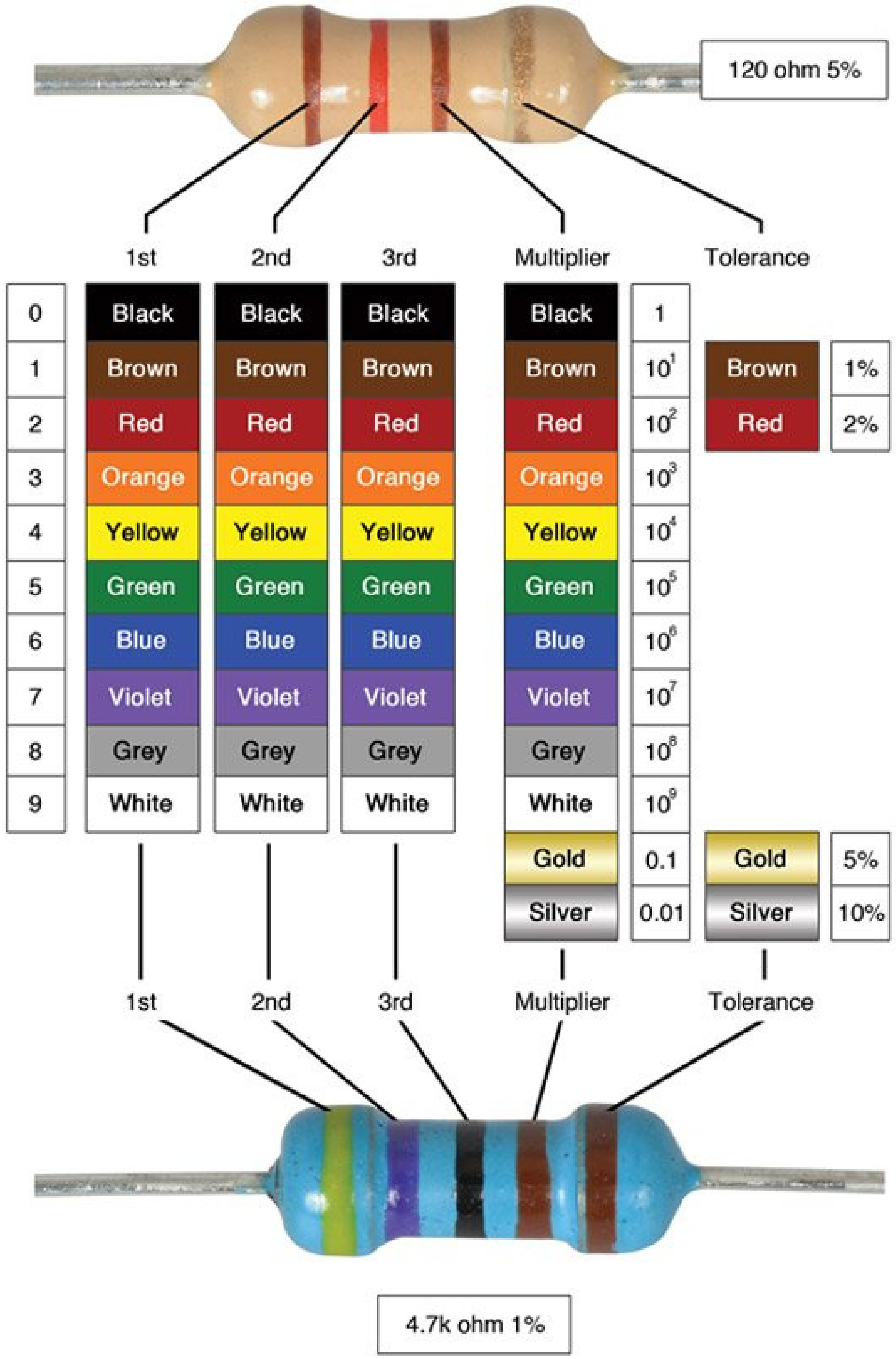
LED: Light Emitting Diode





- A diode that emits light.
- Only allow current to flow from anode to cathode, not the other way around. (One-direction)
- The longer end is the Anode (+) whereas the shorter end is Cathode (-).
- Max current rating: 20mA **(IMPORTANT)**

Resistors



	1st	2nd	3rd	Multiplier		Tolerance	
0	Black	Black	Black	Black	1		
1	Brown	Brown	Brown	Brown	10^1	Brown	1%
2	Red	Red	Red	Red	10^2	Red	2%
3	Orange	Orange	Orange	Orange	10^3		
4	Yellow	Yellow	Yellow	Yellow	10^4		
5	Green	Green	Green	Green	10^5		
6	Blue	Blue	Blue	Blue	10^6		
7	Violet	Violet	Violet	Violet	10^7		
8	Grey	Grey	Grey	Grey	10^8		
9	White	White	White	White	10^9		
				Gold	0.1	Gold	5%
				Silver	0.01	Silver	10%

- Resistors, like the name implied, provide resistance to the circuit.

- Resistors are used to reduce the current flow in the circuit, as some components will break when there is a large current flowing through it. (Eg. LED has a rating of 20 mA)

- Usually in circuits try to use resistors to reduce current flow (IMPORTANT especially LED as has a current rating of 20 mA.)

Types of Sensors

Potentiometer

A variable resistor that changes its resistance based on the **position** of the knob.

Light

A photoresistor that changes its value based on the amount of light it is exposed to. The **more light** there is, the **lower** the **resistance**.

Temperature

Changes its value based on the surrounding temperature. Produces an analog output that is proportional to the temperature detected.

Ultrasonic

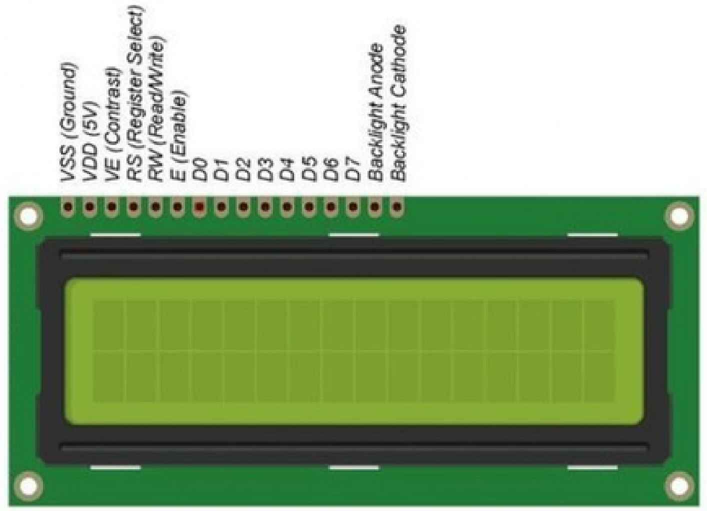
Uses **sound** to detect the distance to an object similar to how bats do. The distance is calculated using the time taken for sound waves to reach the object and bounce back.

There are many more sensors other than these to tinker with!

Types of Actuators

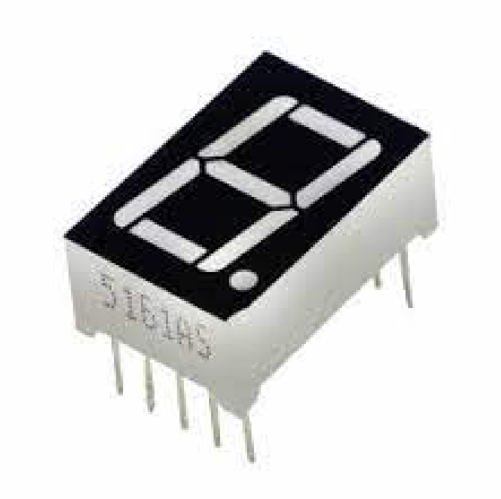
Motors

Has a shaft that rotates when powered. Can be used for wheels and robotic arms.



LCD

A panel that can display text set by the user.



7-segment Display

Contains LED in each segment and can be used to display numbers from 0 to 9 with a decimal point.

Buzzer

A device that produces sound when given an input signal. The tone of the sound produced can be changed based on the input signal.

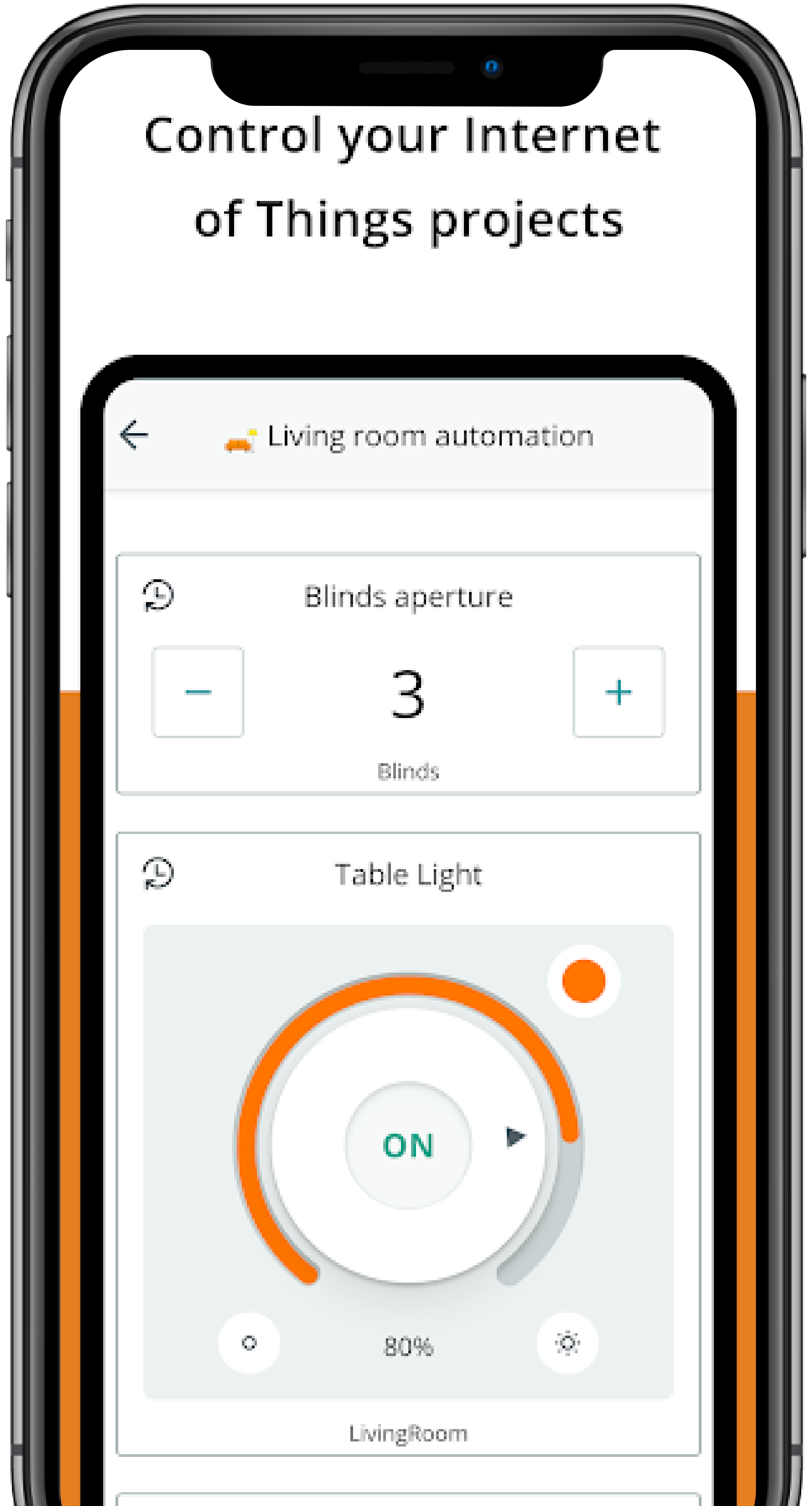
There are also many more actuators other than these to tinker with!

Internet of Things (IoT)

Arduino can be used for IoT

Different devices can be connected together to exchange data over WIFI, Bluetooth and other wireless connections.

Additional WIFI module might be needed for Arduino to be used for IoT as the base Arduino board might not include one depending on the model.



Control your Internet of Things projects

Free Resources

If you need help with your Arduino projects

Arduino Forum

https://forum.arduino.cc/

Arduino Reference

https://www.arduino.cc/reference/en/

Arduino StackExchange

https://arduino.stackexchange.com/

SparkFun Arduino Resources

https://learn.sparkfun.com/resources/tags/arduino?page=all

THANK YOU

Now it is time to start tinkering!